



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/818,572	03/28/2001	Richard Donhauser	Q63564	3655

7590 09/08/2004

SUGHRUE, MION, ZINN, MACPEAK & SEAS, PLLC
2100 PENNSYLVANIA AVENUE, N.W.
WASHINGTON, DC 20037-3213

EXAMINER

LOHN, JOSHUA A

ART UNIT PAPER NUMBER

2114

DATE MAILED: 09/08/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No. 09/818,572	Applicant(s) DONHAUSER ET AL.	
	Examiner Joshua A Lohn	Art Unit 2114	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --
Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 08 July 2004.
- 2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-30 is/are pending in the application.
 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-5, 8, 9, 11, 13-19, 22, 23, 25 and 27-30 is/are rejected.
- 7) ☒ Claim(s) 6, 7, 10, 12, 20, 21, 24 and 26 is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 28 March 2001 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.
 Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
 Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☒ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
 a) ☒ All b) ☐ Some * c) ☐ None of:
1. ☒ Certified copies of the priority documents have been received.
2. ☐ Certified copies of the priority documents have been received in Application No. _____.
3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
- * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|--|---|
| 1) <input type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152) |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____ |

FINAL REJECTION

Response to Arguments

Applicant's arguments filed 7/8/2004 have been fully considered but they are not persuasive.

In response to applicant's submission that Netzer fails to disclose a "message", and teaches away from a message by using a known memory location, the examiner respectfully disagrees. Applicant states that the memory location evaluation takes place at the window boundary and no memory address would be needed to setup a memory access trace file, however the teaching of Netzer uses the window boundary only as a starting position for each code segment that is "evaluated to determine the unique data to be stored" (Netzer, col. 12, lines 26-34). It is this unique data address that is then needed to be provided to the trace file and monitored with that system. It is inherent in the operation of the memory access trace file and system call trace file that some form of message is transmitted, in the electronic system in which they operate, to indicate to the trace files all necessary information regarding the unique segments to be monitored.

In response to the applicant's arguments that a setting up of the trace file is not inherent, and lacks technical reasoning, the examiner respectfully disagrees. Netzer discloses a system of analyzing code segments and associated memory locations to determine unique information segments. Netzer further discloses that these unique locations are monitored by tracing memory accesses and system calls (Netzer, col. 12, lines 20-34). It is true that Netzer does not explicitly describe a process for initializing the trace files, however it would have been inherent to one skilled in the art that some form of electronic message would have been transmitted to provide

Art Unit: 2114

knowledge of what unique data areas must be monitored. This message would have to include a memory address to enable the memory access trace file to monitor the correct unique location, the message would also have had to include some reference to the code related to that memory in order to indicate the system calls that should be traced in relation to the unique memory location. The existence of these trace files and the indicated ability of them to dynamically monitor a detected unique memory area make some form of setting up messaging a requirement for technical feasibility and implementation.

In response to the applicant's argument that the examiner's use of interpretation relies upon hindsight, the examiner respectfully disagrees. The examiner is merely indicating that although the setting up of the trace files does not explicitly teach of including a message the necessary steps for setting up the trace files would be functionally equivalent to the broadest, reasonable interpretation of the limitation of the request message as set forth in the claim.

In light of the above response the examiner maintains the rejection of the claims as reiterated below.

Claim Rejections - 35 USC § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

Claim 1-4, 5, 8, 9, 13, 16-19, 22, 23, 27, and 30 are rejected under 35 U.S.C. 102(e) as being anticipated by Netzer, United States Patent 5,870,607, filed September 11, 1996.

As per claim 1, Netzer teaches executing an application program in an automation device, see column 10, lines 31-32, where the device executing the program is interpreted as being an automation device for automatically executing the program. Netzer also teaches of selecting at least one of a plurality of data addresses to monitor during execution of the application program, see column 11, lines 7-11, where the essential memory locations, with corresponding data addresses, are monitored. Netzer also teaches selecting a code address corresponding to the at least one of a plurality of data addresses selected for monitoring, this is shown in column 11, lines 1-11, where the code corresponding to the monitored data is also selected and analyzed to determine the importance of the related data. Netzer teaches of transmitting to the automation device, as part of a request message the data addresses selected for monitoring and the corresponding code address, this is shown in column 12, lines 20-35, where the request message is interpreted to be the means used to setup the trace file of Netzer, and this setup would inherently include the memory address and related code to allow for a repeatable trace file of memory accesses. Netzer discloses recording the content of the data addresses selected for monitoring when the corresponding code address is reached during execution of the application program, see column 12, lines 20-25, where the values of the memory locations are buffered in storage. Netzer finally teaches of transmitting as part of a result message, the recorded content of the data addresses selected for monitoring, see column 12, lines 26-35, where the result message includes all data relating to the data address to be monitored.

As per claim 2, it is inherent that an application program controls and/or monitors an external process, because all application programs in execution influence and control processes external to the application program, such as the function of the memory, processor, and any peripherals utilized by the application.

As per claim 3, Netzer teaches the code address comprises an address of the application program in proximity to a segment of the application program that influences the content of the data addresses selected for monitoring, this is shown in fact that only code that influences the content of the monitored data address is referenced, and any reference to a code would include a means for addressing that portion of the application which would be included in the tracing of memory accesses, see column 11, lines 1-11.

As per claim 4, Netzer teaches that all memory areas existing in the automation device, including registers, can be referenced using the plurality of data addresses, this is shown in column 6, lines 13-15 and 38-43, as well as column 9, lines 32-42, where Netzer discloses the CPU containing registers, a need to emulate any data flow necessary to recreate a particular state of the CPU, and using the memory access trace file to contain all memory accesses that are required to fully emulate the system operation, which should include all register memory accesses, also see column 23, lines 15-17.

As per claim 5, Netzer teaches recording the data addresses selected for monitoring when the corresponding code address is reached, if a trigger condition is met, where the trigger condition is an indicator of which memory accesses, of the monitored memory locations, need to be stored, see column 10, lines 22-44.

As per claim 8, Netzer teaches that the data addresses selected for monitoring are recorded by the automation device, see the buffering of column 12, lines 20-34, and during execution of the application program are transmitted to a programming device at predetermined events, a predetermined event being a window boundary at which values are saved in a trace file, where the programming device can be a separate processor, see column 20, lines 52-55, where the state machines responsible for monitoring memory access and storing traces are executed on a secondary CPU.

As per claim 9, Netzer teaches of repeated execution of windows within the application program, each window represents a cycle of a cyclically repeating program, and the predetermined event is the end of a window cycle, see column 9, line 14 through column 10, line 53.

As per claim 13, Netzer teaches recording the data addresses selected for monitoring when the corresponding code address is reached, if a trigger condition is met, where the trigger condition is an indicator of which memory accesses, of the monitored memory locations, need to be stored, see column 10, lines 22-44.

As per claim 16, Netzer teaches of an automation device which is used to execute a program, see column 10, lines 31-32, where the device executing the program is interpreted as being an automation device for automatically executing the program. Netzer teaches of sensors providing data to the automation device, these sensors taking the form of the monitoring code for analyzing memory accesses, see column 10, lines 22-28. Netzer teaches of a programming device, in the form of a second CPU used to control some instrumentation, see column 20, lines 48-57. It is inherent that a communications program that enables the automation device and the

Art Unit: 2114

programming device to communicate with one another exists. This is inherent because the second processor is in communication with the target program of the automation device, this mutual contact would require a communications program to exist in some form. Netzer teaches of a computer readable medium for each device on which is stored computer programs for operating and monitoring the automation system, see column 6, lines 18-21, where the typical embodiment of the system has all programs stored in memory, which is a computer readable medium. Netzer also teaches of selecting at least one of a plurality of data addresses to monitor during execution of the application program, see column 11, lines 7-11, where the essential memory locations, with corresponding data addresses, are monitored. Netzer also teaches selecting a code address corresponding to the at least one of a plurality of data addresses selected for monitoring, this is shown in column 11, lines 1-11, where the code corresponding to the monitored data is also selected and analyzed to determine the importance of the related data. Netzer teaches of transmitting to the automation device, as part of a request message the data addresses selected for monitoring and the corresponding code address, this is shown in column 12, lines 20-35, where the request message is interpreted to be the means used to setup the trace file of Netzer and this setup would inherently include the memory address and related code to allow for a repeatable trace file of memory accesses. Netzer discloses executing the application program in the automation device to control and/or monitor an external process, see column 10, lines 31-32, and it is also inherent that an application program controls and/or monitors an external process, because all application programs in execution influence and control processes external to the application program, such as the function of the memory, processor, and any peripherals utilized by the application. Netzer discloses recording the content of the data

Art Unit: 2114

addresses selected for monitoring when the corresponding code address is reached during execution of the application program, see column 12, lines 20-25, where the values of the memory locations are buffered in storage. Netzer finally teaches of transmitting as part of a result message, the recorded content of the data addresses selected for monitoring, see column 12, lines 26-35, where the result message includes all data relating to the data address to be monitored.

As per claim 17, Netzer teaches the code address comprises an address of the application program in proximity to a segment of the application program that influences the content of the data addresses selected for monitoring, this is shown in fact that only code that influences the content of the monitored data address is referenced, and any reference to a code would include a means for addressing that portion of the application which would be included in the tracing of memory accesses, see column 11, lines 1-11.

As per claim 18, Netzer teaches that all memory areas existing in the automation device, including registers, can be referenced using the plurality of data addresses, this is shown in column 6, lines 13-15 and 38-43, as well as column 9, lines 32-42, where Netzer discloses the CPU containing registers, a need to emulate any data flow necessary to recreate a particular state of the CPU, and using the memory access trace file to contain all memory accesses that are required to fully emulate the system operation, which should include all register memory accesses, also see column 23, lines 15-17.

As per claim 19, Netzer teaches recording the data addresses selected for monitoring when the corresponding code address is reached, if a trigger condition is met, where the trigger

Art Unit: 2114

condition is in indicator of which memory accesses, of the monitored memory locations, need to be stored, see column 10, lines 22-44.

As per claim 22, Netzer teaches that the data addresses selected for monitoring are recorded by the automation device, see the buffering of column 12, lines 20-34, and during execution of the application program are transmitted to a programming device at predetermined events, a predetermined event being a window boundary at which values are saved in a trace file, where the programming device can be a separate processor, see column 20, lines 52-55, where the state machines responsible for monitoring memory access and storing traces are executed on a secondary CPU.

As per claim 23, Netzer teaches of repeated execution of windows within the application program, each window represents a cycle of a cyclically repeating program, and the predetermined event is the end of a window cycle, see column 9, line 14 through column 10, line 53.

As per claim 27, Netzer teaches recording the data addresses selected for monitoring when the corresponding code address is reached, if a trigger condition is met, where the trigger condition is in indicator of which memory accesses, of the monitored memory locations, need to be stored, see column 10, lines 22-44.

As per claim 30, Netzer teaches of means for displaying or outputting transmitted and recorded data in the form of the memory access trace file, which is output to, see column 12, lines 20-34.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

Claims 11 and 25 are rejected under 35 U.S.C. 103(a) as being unpatentable over Netzer in view of Testardi, United States Patent number 6,249,882, filed June 15, 1998.

As per claim 11, Netzer does not teach that instructions required for monitoring can be masked in the application program.

Testardi discloses masking, through use of comments, the test code, which acts to monitor for errors, in the application program, see column 5, line 58 through column 6, line, 12.

It would have been obvious to use the masking technique of Testardi in the testing and development system of Netzer.

This would have been obvious because Netzer teaches of adding instrumentation code to the original computer program to allow for through testing and creation of trace files, see column 3, lines 31-48. Netzer does not provide a course of action to allow for operation of the program in a non-testing environment. Testardi teaches of a system that allows for the operation of the program to not be impacted by the inclusion of test code when the code is not in use, see column 6, lines 9-12. It is well known in the art that eventually a software program will be substantially free of errors, see column 1, lines 18-21 of Testardi, and no longer need to operate in a test environment. It would have been obvious to use the commenting method of Testardi to mask the

Art Unit: 2114

monitoring code of Netzer and allow for the program code of Netzer to be executed in a system without being inhibited by the monitoring code.

As per claim 25, this claim contains the same additional limitations as claim 11, and is rejected under the grounds mentioned previously.

Claims 14, 15, 28, and 29 are rejected under 35 U.S.C. 103(a) as being unpatentable over Netzer in view of Silva et al., United States Patent number 6,163,805, filed October 7, 1997

As per claim 14, Netzer discloses a system in which a request message is sent to the automation device, as part of a request message the data addresses selected for monitoring and the corresponding code address, this is shown in column 12, lines 20-35, where the request message is interpreted to be the means used to setup the trace file of Netzer and this setup would inherently include the memory address and related code to allow for a repeatable trace file of memory accesses. Netzer does not disclose an acknowledgement message and a job number from the automation device to a programming device acknowledging receipt of the request.

Silva discloses an acknowledgement message and a job number from a test requester and a test executor, see column 14, lines 14-28.

It would have been obvious to one skilled in the art at the time the invention was made to include the acknowledgement message of Silva in the invention of Netzer.

This would have been obvious because Netzer teaches of the programming device taking the form of a second processor, see column 20, lines 50-57. This programming device is responsible for determining which memory addresses should be monitored and communicating this information to the processor responsible for executing the test application. The

communication of these two processors is not described in detail. Silva discloses a means for communicating between processing devices in a multi-processor system executing test jobs. The programming device, or second processor of Netzer, is the same as the requester of Silva and the automation device, or program execution processor of Netzer, includes the functions of the dispatcher of Silva. The use of job acknowledgements as described by Silva would obviously improve the effectiveness of the communication between the processors of Netzer by providing indicators of the status of execution, see Silva column 14, lines 18-20.

As per claim 15, Netzer fails to teach of sending a release message from a programming device to the automation device to end monitoring.

Silva teaches of a release message that allows the programming device to end testing, see column 10, lines 28-35, where a lack of jobs results in the "execute-nothing" message being sent to the automation device.

It would have been obvious to one skilled in the art at the time of the invention include the "execute-nothing" message in the environment of Netzer.

This would have been obvious because Netzer shows a strong desire to minimize setup and replay times when executing test programs, see column 6, lines 54-56. The use of the "execute-nothing" message disclosed by Silva would allow replay to be paused until the programming device provides memory address monitoring jobs. It would have been obvious that if replay is continued without knowing the desired addresses to monitor, the system would have wasted time in clearing out any information collected when an address monitoring job is finally provided. It would have been obvious to use the "execute-nothing" command of Silva to release

Art Unit: 2114

the automation device as taught by Netzer to allow it to be in a ready state for any new replay window and address monitoring requirements.

As per claim 28, this claim contains the same additional limitations as claim 14, and is rejected under the grounds mentioned previously.

As per claim 29, this claim contains the same additional limitations as claim 15, and is rejected under the grounds mentioned previously.

Allowable Subject Matter

Claims 6, 7, 10, 12, 20, 21, 24, and 26 are objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims.

Conclusion

THIS ACTION IS MADE FINAL. Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.


Art Unit: 2114

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Joshua A Lohn. The examiner can normally be reached by telephone at (703) 305-3188 until October 15, 2004, at which point the examiner should be contacted at telephone number (571) 272-3661. The examiner is usually in office M-F 8-4.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Robert Beausoleil can be reached on (703) 305-9713. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

JAL


SCOTT BADERMAN
PRIMARY EXAMINER